International Journal of Scientific Engineering and Technology
Volume No.4 Issue No.8, pp: 447-450
ISSN :2277-1581
01 August 2015

Conversion of English Text-to-Speech (TTS) Using Indian Speech Signal

[1]R.Shantha Selva Kumari, [2]R.Sangeetha
Dept. of ECE, Mepco Schlenk Engg. College ,Sivakasi, India
[1]rshantha@mepcoeng.ac.in, [2]rsangeetha121@gmail.com

Abstract: The objective of this paper is to convert the english text into speech. The conversion of english text into speech is done by using a stored speech signal data. Text to speech conversion module is designed by the use of matlab. By the use of microphone the phonemes (alphabets, numbers, words) are recorded using a goldwave software. The recorded .wav (sounds) files are saved as a database separately. The phonemes are extracted from the text file. For text to speech conversion the concatenation method is proposed. The recorded speech are concatenated together to produce the synthesized speech. The resulting speech output is assessed by listening test. The Mean Opinion Score (MOS) value is calculated for the synthesized speech output as the performance measure. In future work, a miniaturized hardware implementation will be developed for helping visually impaired persons in understanding text they come across in day today life.

Keywords: Speech signal, goldwave, phonemes, Text-to-Speech (TTS), Mean Opinion Score (MOS).

I. INTRODUCTION

Language is the ability to communicate one's thoughts by means of a set of signs (text), gestures, and sounds. Speech is most widely used for communication between people.

Text to speech (TTS) conversion [1] is the process of converting information stored as a data or text into speech. It is useful for blind people as audio reading device. There are many speech synthesizers using complex neural network design.

The important qualities of speech synthesis systems are naturalness and intelligibility. Naturalness describes the output speech sounds similar to human speech and intelligibility is which the output speech sound is understood. There are two types of synthesis speech generations are available, concatenative synthesis and formant synthesis [2].

Concatenative synthesis is based on the concatenation of segments of recorded speech. Connecting the prerecorded natural utterances [3] is the better way to produce understandable and natural sounding synthesis speech. However, concatenative synthesizers are requires more memory capacity. The most important aspect in concatenative synthesis is to find correct unit length. The selection is trade-off between longer and shorter units. With longer unit high naturalness, less concatenation points are achieved, but the amount of required units and memory is increased. The shorter units less memory are needed, but the sample collection and labeling procedures become difficult and complex. The present systems units used are usually words, syllables and phonemes.

Formant synthesis does not use human speech samples at runtime. The synthesized speech output is created using adaptive synthesis and an acoustic model. This method is also called as rules based synthesis. Formant based synthesizer employs demi-syllable concatenation, involves identifying and extracting the formats from an actual speech signal and then using this information to construct demi-syllable segments each represented by a set of filter parameters and a source signal waveform. The basic unit being demi-syllable requires numerous concatenation and hence the speech lacks from continous flow.

The concatenative synthesis approach is used in the proposed Text-to-Speech (TTS) conversion system, where the natural speech output is concatenated to give the resultant speech output. Based on the input, the phoneme(.wav) are selected from the database and concatenated [4],[5] using MATLAB to generate the output speech.

The application of TTS system are telecommunication service, language education, vocal monitoring, aid for handicapped persons, vocal monitoring and mainly for visually impaired persons.

II. PROPOSED SYSTEM

The block diagram of the proposed system is shown in Fig.1.The concatenative synthesis approach is used in the proposed TTS system.

A. Concatenative Synthesis

In concatenative speech synthesis system there are three sub-types are available. They are namely Unit selection synthesis [6]-[10], Diphone synthesis and Domain specific synthesis. In our work, we used Domain specific synthesis in which , the prerecorded words are stored with corresponding names and maintained in a database. Based on the input text, the phonemes (.wav) are selected from the database and concatenated using MATLAB to create the resultant output.

B. Domain Specific Synthesis

Concatenation of words is performed by domain specific synthesis. It concatenates prerecorded words and phrases to create complete utterances. It is very simple to implement and in viable use for a long time. The domain specific synthesis systems are having limited by the words and phrases in their database so the naturalness of these systems can be high because the varieties of sentence types are limited.

C. Creation of Database

Phonemes are probably the most commonly used units in the speech synthesis because they are the normal linguistic presentation of speech.

There are different factors to be considered while designing a TTS system that will produce understandable speech. The first step in the design of TTS system is to select the most suitable units or segments of speech that results in smooth utterance.

Building the units list consists of three main phases. First, the natural speech must be recorded so that all used units (phonemes) within all possible contexts are included. After this,

ISSN :2277-1581
01 August 2015

the units must be labeled from spoken speech data, and finally, the most suitable units must be chosen. Gathering the samples from natural speech is usually very time-consuming. The implementation of rules to select correct samples for concatenation must also be done very carefully.

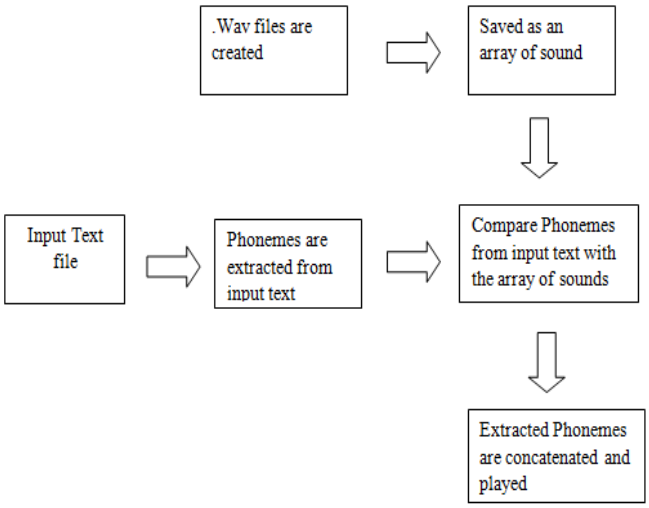

Fig. 1. Block diagram of proposed system

The voice which is recorded manually contains some delay. This causes a greater time drop between two repeated utterances. This makes the speech a bit unpleasant and unnatural to listen. Hence there is a need to remove this delay.

D. Implementation Algorithm

1) Character – To – Tone

Let us start text to speech synthesis with a simple character to voice conversion. The database required for character to voice conversion is recorded alphabets (a-z, A-Z), digits (0-9) in the form of wave files (.wav).

The next step in converting text to speech is to create a text file (.txt). Once the file is created, it is opened and read in MATLAB.

In MATLAB all the data is stored in the form of a matrix. For every element read, corresponding wave file is played so as to output the sound of that character.

Algorithm:

STEP 1: Create a database of various wave files.
STEP 2: Create a text file (.txt).
STEP 3: Open the .txt file in MATLAB.
STEP 4: Read the file opened.
STEP 5: For each and every Character read and play corresponding wave (.wav) file.

But as said earlier, there exist some delay by default while recording a sound. This delay has to be removed to get a continuous utterance of speech.

2) Word –To – Speech

Character to voice is not a big task. This is because there are only 26 characters in English and each character has a unique pronunciation. However when we have to read lengthy texts, character to voice is not recommended at the user level, as it is difficult to make out a word from the characters read.

As we have played the wave files corresponding to every character read, in character to voice conversion, we can also play the wave files for every word read. To record all the words of a dictionary, the database memory also increased. Hence choosing sound unit with proper length is important, so that the word is natural and understandable when synthesized. Once the text is read, for every word the corresponding wave files are concatenated and played.

Algorithm:

STEP 1: Create a database of various words.
STEP 2: Create the text file (.txt).
STEP 3: Open the .txt file in MATLAB.
STEP 4: Read the file.
STEP 5: Concatenate the .wav files accordingly and play them.
By using the above algorithm the words are concatenated and played accordingly. The text file contains number of lines.
Line by line the words are concatenated and played.

III. Results And Discussion

A. Character –To – Tone

Input Text: HAPPY

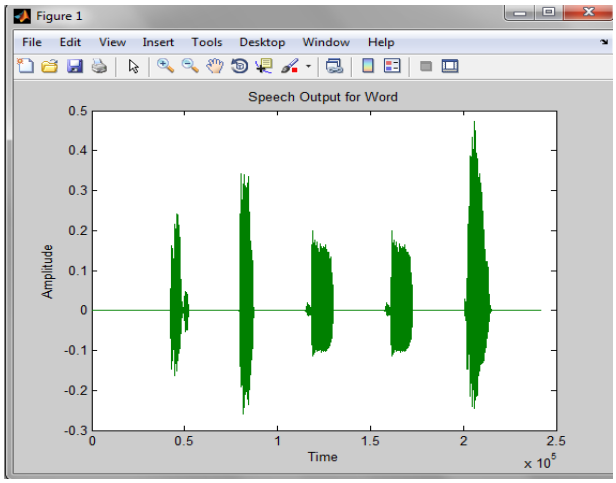

Fig.2. Plot of the sound ‘H’, ‘A’, ‘P’, ‘P’, ‘Y’.

The text has the input as HAPPY. The Fig.2. shows the each and every character of the input text reads and the corresponding wave file is played. During Character to voice conversion, it is observed that the delays are removed, the natural the output sounds.

B. Word – To –Speech

Input Text: Have a Nice Day

International Journal of Scientific Engineering and Technology
Volume No.4 Issue No.8, pp: 447-450

ISSN :2277-1581
01 August 2015

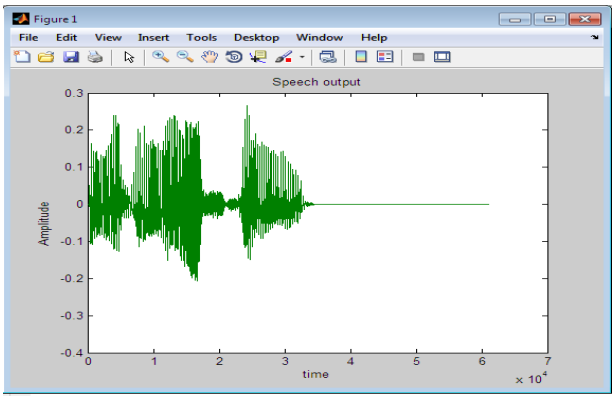

Fig.3. Voice Output for the input sentence

As the first step to play the word, we started playing individual sounds of the word separately and then concatenated them to form a sentence. The above fig .3. shows the text to speech output of the sentence "Have a Nice Day".

C. Input Text File

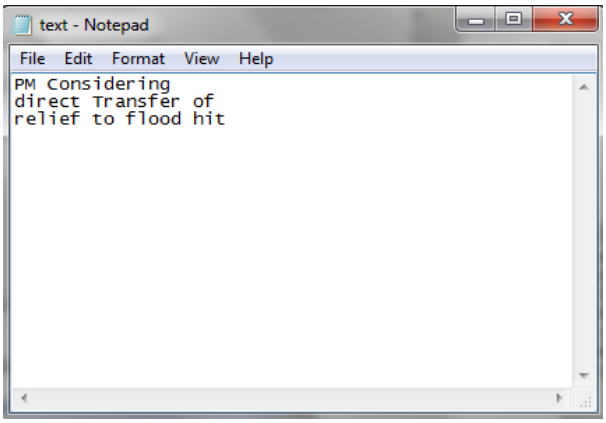

Fig.4. Input Text File

The above text file is given as the input of the TTS system. The fig.4. text file contains three lines of text to converting into speech.

D. Text to Speech output for the text file

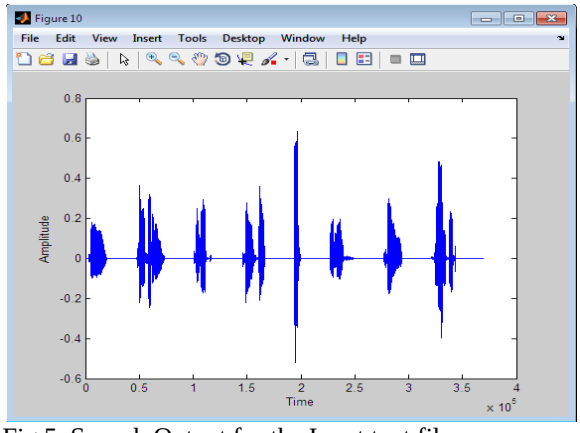

Fig.5. Speech Output for the Input text file

The text file is given to the input for the TTS system. The text file contains The fig.5. shows the speech output for the text file. The individual sounds of the words are separately played and then concatenated them to form a sentence.

E. Performance Measure

To evaluate the quality of the speech produced by the developed system we passed out proper listening tests. Testing is carried out using subjective test. In subjective tests, individual listeners listen to and rate the heard audio quality of test sentences by both male and female speakers over the communications medium being tested.

Mean Opinion Score is composed of five scores of subjective quality, 1-Bad, 2-Poor, 3-Fair, 4-Good, 5-Excellent. The MOS score of a certain vocoder is the average of all the ranks voted by different listeners of the different voice file used in the experiment, here tests are conducted with twenty students. The tests were conducted in the laboratory environment by playing the speech signals through headphones. Then they were asked to judge the distortion and quality of the speech.

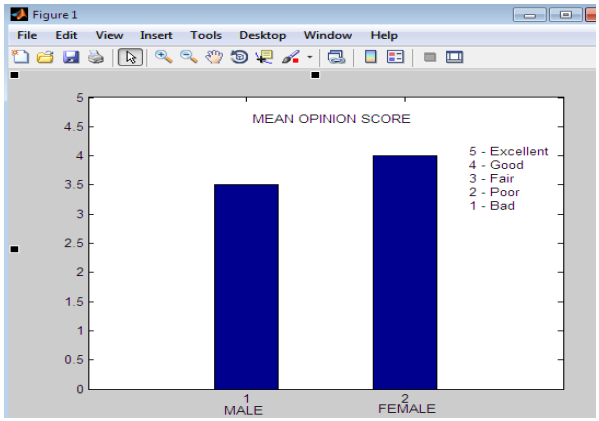

Fig.6. MOS value for generated speech output for the text file

The Mean Opinion Score (MOS) value is calculated for the generated concatenative speech output for the newspaper image as the input of the OCR system. The fig 6.17 shows the MOS value for male and female speech output. For female speech output the MOS value is good.

IV. CONCLUSION

The proposed system gives a very simple approach for text to speech conversion. The limitations considered are, (i)Concatenating the sounds depending on the words, (ii) Sound quality and naturalness of the output speech. The Text to Speech (TTS) conversion is performed using alphabets, numbers and words. The database contains phonemes are limited. For increasing the number of phonemes (.wav) the memory size also increases. The algorithm is checked for both male and female. The performance measure for the concatenative speech is calculated in terms of Mean Opinion Score (MOS). The MOS value for female is 4 and for male is 3.5.The delay is reduced and mismatching of words also reduced. In future work a miniaturized hardware implementation will be developed for helping visually impaired persons in understanding text they come across in day to day life.

References

i. *Chucai Yi, Yingi Tian, K.Anuradha,"Text to Speech Conversion," IEEE Transaction on vol.19,pp .269-278, 2013.*

ii. *R.R. Itkarkar, D.T.Mane, S.D.Suryawanshi, Manoj Kumar Singh, "High Quality Text to Speech Synthesizer using Phonetic Integration" IJARECE 2014, 133-136.*

ISSN :2277-1581
01 August 2015

iii. Haojin Yang, Hasso-Plattner, C.Meinel, "Design of Multilingual Speech Synthesis System," Intelligent Information Management, pp. 58-64, 2010.

iv. Francesc Alias, Xavier Sevillano, Joan Claudi Socoro, Xavier Gonzalvo, "Towards High Quality Next Generation Text-to-Speech Synthesis: A Multidomain Approach by Automatic Domain Classification," IEEE Transaction on Audio, Speech and Language Processing vol.16, No.7, pp. 1340-1354, 2008.

v. Vijay Laxmi Sahu, Babita Kubde, "Design and Development of a Text-To-Speech Synthesizer for Indian Language:" A Review International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064 Volume 2 Issue 1, January 2013.

vi. Gurpreet Singh, Chandan Jyoti Kumar, Rajneesh Rani, Dr. RenuDhir, "Building HMM based Unit Selection Speech Synthesis System," IJARCSSE Volume 3, Issue 1, pp 257-263, January 2013.

vii. Chirag I Patel, Ripal Patel, Palak Patel, " Integrated Automatic Expression Prediction and Speech Synthesis" International Journal of Scientific & Engineering Research, Volume 2, Issue 5, May-2011 .

viii. A. F. Mollah, S. Basu, M. Nasipuri, "Robust Speaker Adaptive HMM based Text to Speech System", International Journal of Computer Science and Applications, 1(1), pp. 33-37, June 2010.

ix. A. F. Mollah, S. Basu, M. Nasipuri and D. K. Basu, "Parameter generation methods with Rich context model for Text to Speech Synthesis", Proc. of the Eighth IAPR International Workshop on Graphics Recognition (GREC`09), pp. 263-270, July, 2009.

x. Diego J. Romero, Leticia M. Seijas, Ana M. Ruedin, "The IBM Expressive Text to Speech Synthesis for American English," JCS&T Vol. 7 No. 1 April 2007.

xi. Shen, H. and Coughlan, "An RNN based Prosodic Information Synthesizer for Mandarin Text to Speech", Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop, 2006.

xii. H.Li, D.Doerman, and O.Kia, "A system for converting English Text into Speech," IEEE Transactions on Image Processing, pp. 147-156, 2004.

xiii. Yu Zhong, Hongjiang Zhang, and Anil K.Jain, "Letter to sound Rules for automatic translation of English Text to Speech ," IEEE Transactions on Pattern Analysis and Machine Intelligence, 22, (4), pp. 384-392, 2000.

xiv. Yassin M.Y.Hassan and Lina J.Karam, "Clusttering of Duration Patterns in Speech for Text to Speech Synthesis," IEEE Transactions on Image Processing, 9(11), pp.1978-1983, 2000.

xv. A.K.Jain, and B.Yu, "Applying a Speaker Dependent Speech Compression Technique to Concatenative Synthesizer," IEEE Transaction on Audio, Speech and Language Processing, 20(3), pp.294-308, 1998.